Gorilla REPL

Welcome to gorilla :-)

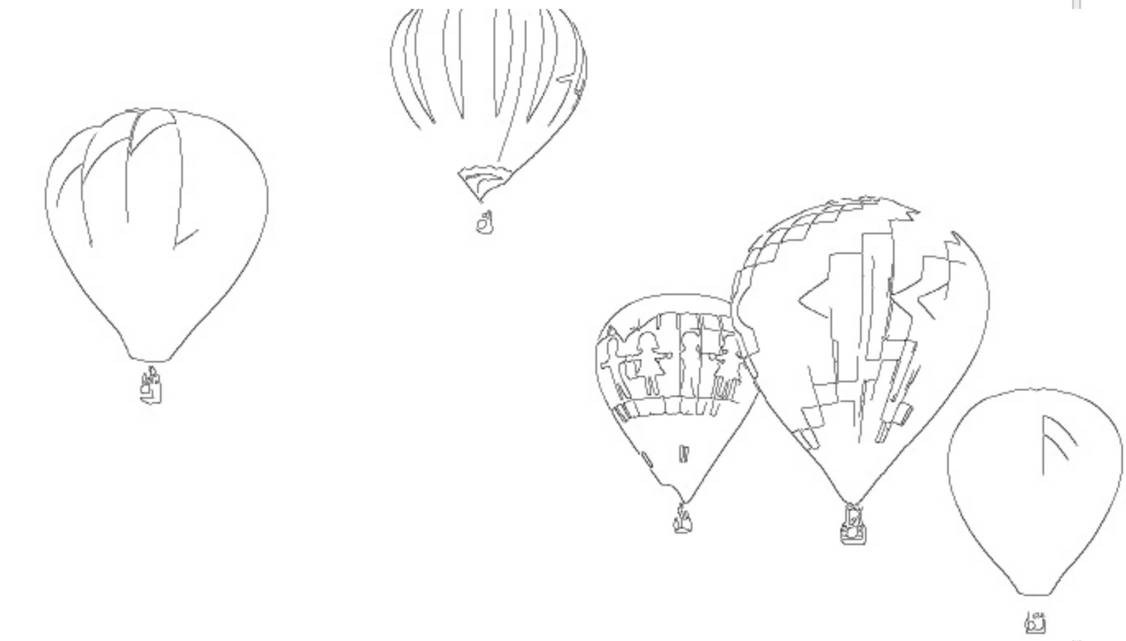
Shift + enter evaluates code. Hit ctrl+g twice in quick succession or click the menu icon (upper-right corner) for more commands ...

It's a good habit to run each worksheet in its own namespace: feel free to use the declaration we've provided below if you'd like.

```
(ns opencv3.dolphin
(:require
  [opencv3.utils :as u]
  [opencv3.core :refer :all]))
```

```
nil
```

```
(->
"http://www.v3wall.com/wallpaper/1366_768/0912/1366_768_20091223010850201138.
jpg"
 u/mat-from-url
 (median-blur! 3)
 (cvt-color! COLOR_BGR2GRAY)
 (canny! 300.0 100.0 3 true)
 (bitwise-not!)
 (u/resize-by 0.5)
 (u/mat-view))
```



```
(def cat
  (-> "resources/images/cat.jpg"
  imread))
```

```
#'opencv3.edges/cat
```

```
(-> cat
  (clone)
  (median-blur! 7)
  (cvt-color! COLOR_BGR2GRAY)
  (laplacian! 0)
;  (bitwise-not!)
  (u/resize-by 0.5)
  (u/mat-view))
```

```
(def headphones (->
    "resources/morph/headphone.png"
    imread
    clone
   (cvt-color! COLOR_BGR2GRAY)))

(def contours (new-arraylist))
(find-contours headphones contours (new-mat) RETR_LIST CHAIN_APPROX_SIMPLE)

(doseq [c contours]
  (if (> (contour-area c) 100 )
    (let [ rect (bounding-rect c)]
      (if (> (.height rect) 28)
      (rectangle
        image-c
        (new-point (.x rect) (.y rect))
        (new-point (+ (.width rect) (.x rect)) (+ (.y rect) (.height rect)))
        (new-scalar 255 100 100)
        3)))))
(u/mat-view image-c)
```